International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Keylogger: An Advanced Method for Computer Monitoring

Pranav Bhalerao[1], Priyanshu Vadhwani[2], Atharva Wagaskar[3], Prof. Sairabanu Pansare[4]

[1,2,3,4,5]Department of CSE, Nutan College of Engineering and Research, Talegaon Dabhade, Maharashtra.

Abstract:
With the help of the increasing reliance on the digital systems,cybersecurity threat and such as keyloggers have evolved in to many cyber attacks and with the help of many sophisticated tools which are capable of teaching their user about privacy and data security. So we are presenting this paper for the implementation of an advanced software-based keylogger designed to operate the stealthily in a windows environment.The system we developed that captures the keystrokes, also captures the time of the keystrokes, and also captures the mouse logging and stores data locally with email-based transmission.The performance of the proposed tool is evacuated based on memory footprint,CPU usage and the detection rate through the multiple security problems. The result also indicate that the implemented keylogger demonstrates low resource usage and effective stealth behaviour, making it a potent surveillance tools.The research not only highlight the technical feasibility of building such systems but also underscores the legal responsibilities and the ethical hacking associated with cybersecurity research. Potential countermeasures and enhancements for detection are also discussed.

Keywords: Keylogger, Cybersecurity, Malware, Keystroke Logging, Mouse Logging, Hacking, Keystrokes, Cyber Threats

INTRODUCTION

In today's digital age, the proliferation of internet-connected devices and software applications has significantly increased the risk of cybersecurity threats.One such threat is, **keyloggers,** also represents a persistent and evolving form of malware designed to monitor and record user activity---keystrokes---without the user;s knowledge.Originally developed for legitimate purposes such as parental monitoring and corporate auditing,keyloggers have been exploited by malicious actors to steal or hack any sensitive data,including credentials,credit cars numbers and personal messages.Keyloggers are operated stealthy.Often bypassing traditional antivirus mechanisms through the use of sophisticated evasion techniques.Their growing complexity also combined with their ability to remain undetected for long periods,makes them a tools for helping cyber criminals.While in numerous studies,they have examined the keylogger behaviour and detection methods,there is also a critical need for practical and real-world demonstration to better understand their implementation and weakness.

This paper presents the designed and implementation of software-based keylogger for windows platforms like operating system,Linux,macOS etc.The system is capable of recording user keystrokes,capturing

screen activity and mouse logging.Through performance evaluation and comparative analysis, this work aims to demonstrate the feasibility and risks associated with such tools,ultimately contributing to the development of more robust detection and prevention.Ethical considerations and responsible also disclosure practices are emphasized throughout,ensuring that the research supports defensive security objectives.

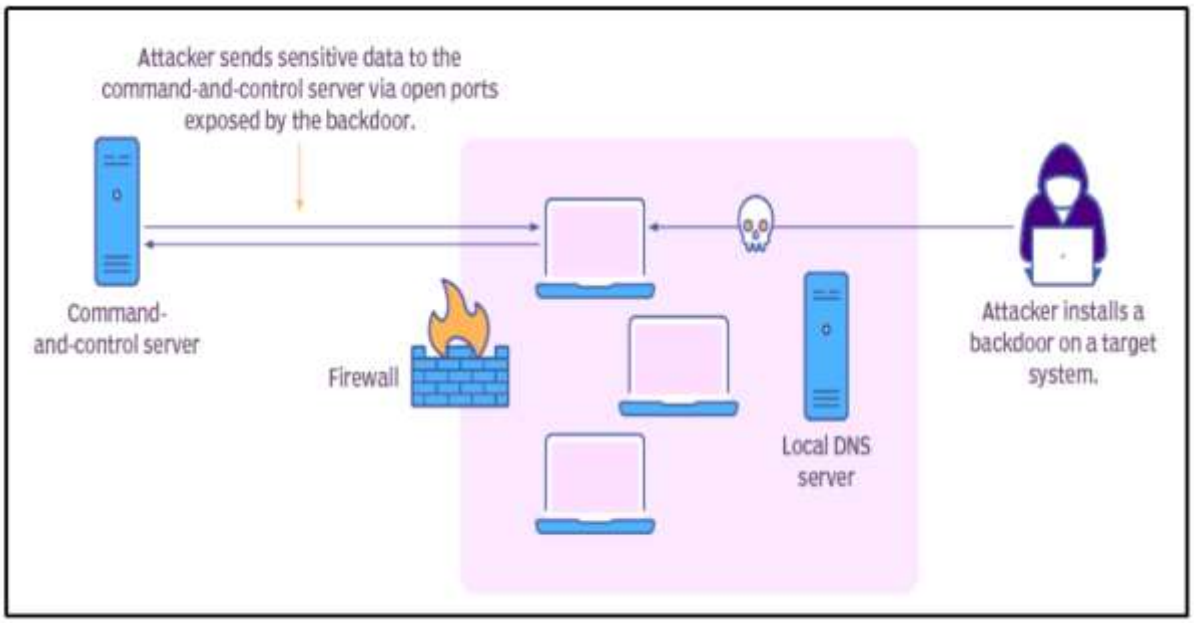

Fig.1. Keylogger Forensic

Fig.2. Keylogger software

LITERATURE REVIEW

The keylogger forensics has witnessed substantial research efforts in recent years due to growing sophistication and persistence of keylogging threats. A variety of detection,analysis and prevention techniques have been proposed and studies by researchers globally.

Case et al.(2020) introduced *Hook Tracer*, a system that uses memory forensics and SetWindowsHookEx monitoring to automatically detect keylogger behaviour.Their approach emphasizes scalability and automation but may fall short in detecting highly or rare techniques due to its dependence on environments.

Tom Olzak(2008) explored both hardware and software based keyloggers, including wireless intercept and acoustic logging methods.While hid findings offered useful insights for legal and illegal uses,the work highlighted the challenges in dealing with hardware-based keyloggers and the need for improved preventive strategies.

Bhardwaj et al.(2020) discussed keyloggers as "silent cybersecurity weapons," identifying a range of techniques from software to acoustic logging. Their research emphasized the potential for both ethical and malicious uses,indicating a need for broader counter measures.

Shah and Priya (2020) performed real-time malware analysis analysis using both static and dynamic approaches. Their findings revealed that while stealth and wide applicability are major advantages of modern keyloggers, they also pose severe risks to user privacy due to their covert nature.

Divedi and Tripathi presented a detailed analysis of stealthy keyloggers and emphasizes their legitimate and illegitimate applications. The study proposed comparative frameworks but also warned against ethical concerns and misuse, indicating a gap in public awareness and detection tools.

Ra him and Saleh proposed a keylogger monitoring system based on exact string ,matching algorithms. While their approach effectively captured keyboard inputs, it lacked the capacity to address more advanced features like virtual or remote keylogging.

Chinchalkar (2024) introduced an innovative detection method using machine learning, specifically combining Support Vector Machine (SVM) with Dendritic Cell Algorithms (DCA). Achieving over 99% accuracy, the approach proved promising but required high-quality datasets for optimal performance.

Singh and Choudhary focused on dynamic, static and traffic analysis for keylogger detection. They highlighted the importance of proactive defense mechanisms but noted challenges in interpreting behaviour due to the increasingly covert operation of modern keyloggers.

Hossain and Rahman (2019) combined memory forensic analysis with network monitoring to detect keylogger activity. Their hybrid model demonstrated broad threat coverage but faced performance overheads and lacked precise accuracy metrics.

Jaiswal and Jana (2022) explored detection methods based on understanding the architecture and functioning of keyloggers. While their work has mainly educatinal, it offered a comprehensive foundation for integrating keylogger topics into academic curriculam.

Kazi et al. (2023) leveraged both static and dynamic analysis techniques for real-time monitoring. Despite notable benefits in educational use and improved detection,their system faced issues like false positives and the rapid evolution of keylogger variants.

Wazid and Katal (2013) developed a honeypot based monitoring framework for keylogger spyware. Although effective in log analysis and threat detection, the method required substantial system resources and was limited in generalization.

Hassan and Maarof (2017) provided a broad survey of keylogger technologies, outlining various detection and analysis approaches. While the study raised awareness and summarized existing methods well, it lacked implementation details and actionable frameworks.

Collectively, these studies underline the growing complexity of keyloggers and the need for advanced, adaptive and ethical countermeasures. However, most research focuses on detection and real-world validation--an area this paper seeks to address by presenting a practical implementation and forensic analysis of a custom keylogger system.

Solairaj et al. (2016) analyzed various software based detection methods and highlighted that most conventional antivirus system fail to detect advanced keyloggers due to their rootkit capabilities. Their work emphasized the need for integrating kernel-level scanning and system-call tracing to improve detection.

Dadkhah et al. (2014) explored the stealth capabilities of keyloggers and demonstrated how undetectable keyloggers evade behavioral and heuristic-based systems through polymorphic and metamorphic transformations. The study showed that testing keyloggers in virtual environments with controlled logging could enhance detction models.

Ladakis et al. (2013) introduced a novel stealthy keylogger based on GPU-level execution. The paper proved that by offloading malicious activity to the GPU. Traditional CPU-centric monitoring systems could be bypassed. This reveled a significant gap in existing detection systems and introduced a new frontier in malware development.

Kuncoro and Kusuma (2018) examined the risls of mobile banking in the presence of keyloggers. They demonstrated how mobile keyloggers, when combined with phishing and screen capturing techniques,pose a substantial threat to financial institutions. This research brought attention to Android based keylogger variants that bypass app-level sand-boxing.

Jvaheri et al.(2018) proposed a kernel level hook detection framework capable of intercepting system calls modified by keyloggers. Their experimental results showed promise in identifying hidden system routines, although the approach faced performance limitations in real-time systems.

Shetty (2005) provided a foundatinal study on spyware-based keyloggers, emphasizing their classification and operational behavior. Though dated, the study remains relevant for understanding early detection strategies and forming the basis for modern anomaly detection algorithms.

Sagiroglu and Canbek (2009) conducted a thorough study on the social engineering aspects of keyloggers. Their work demonstrate that many users unknowinglu install keyloggers via seemingly harmless applications or emails. The psychological element of attacks was emphasized as an overlooked dimension in forensic investigations.

SYSTEM ARCHITECTURE

1. **Data Acquisition Layer -** It utilizes forensic imaging tools like FTK Imager and Dumplt tp capyure both volatile(RAM) and-volatile (disk) memory. It also ensures bit-by-bit integrity while preserving forensic soundness.
2. **Memory and Disk Analysis Layer -** Tools such as Volatility Framework and Red line are employed to analyze memory dumps for hidden or running keylogger processes.The disk imaging tools also replicate entire storage devices for post event analysis.
3. **Networking Monitoring Module -** Wire shark and Snort analyze real-time traffic and log suspicious communication between the host and external command-and-control (C2) servers.It also focuses on

identifying keystrokes ex-filtration patterns.

4. **Behavioral Monitoring Layer -** Uses tools like Process Monitor (ProcMon) and internals Suite to track registry changes,file creation or unusual access attempts.
5. **Detection Engine -** Employs Machine Learning algorithms such as SVM and Anomaly dection to distinguish between malicious behaviors. It also integrates known IOC(Indicators of Compromise) patterns and heuristics.
6. **Logging and Reporting Module -** Generates comprehensive reports documenting artofacts like keystroke logs, process IDs, system timelines and file hashes. It also supports export to legal or investigative formats(e.g.,PDF/CSV).

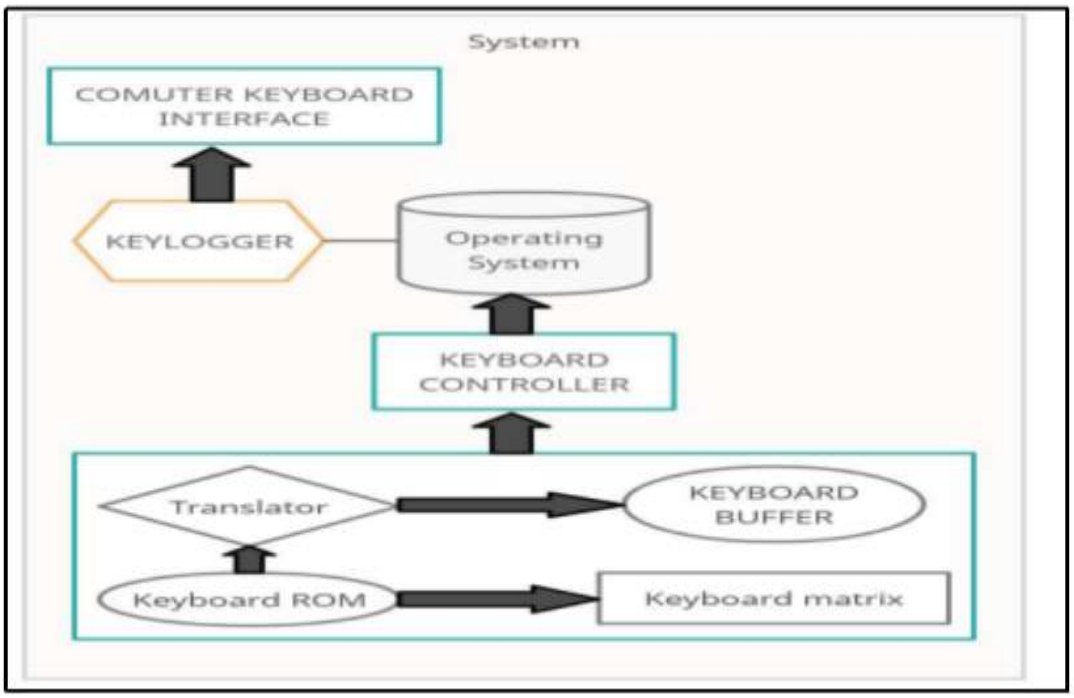

Fig.2. The keylogger's Block Diagram

The system architecture the keylogger between the keyboard interface and the operating system, enabling it to intercept user at an early stage. By accessing the keyboard buffer and controller via low-level hooks, the keylogger remains undetectable to standard user-mode antivirus tools.

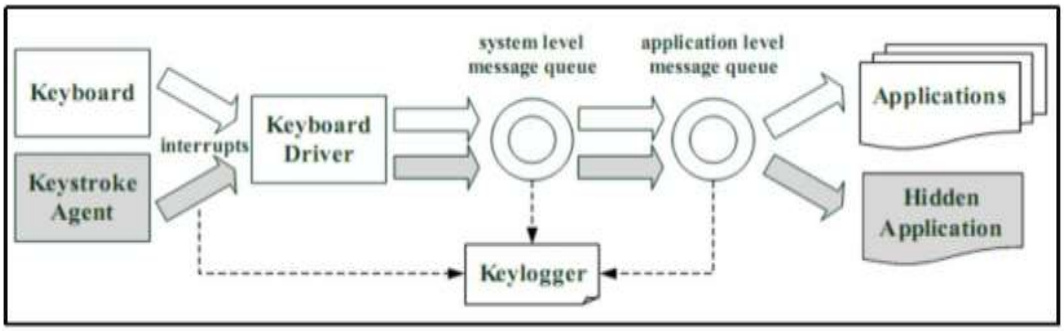

Fig.3.Class Diagram

1. Keyboard and Keystroke Agent - The system begins at the keyboard, which captures physical key presses. The keystroke agent processes keyboard interrupts and forwards signals to the keyboard driver. This agent represents the low-level interception mechanism.
2. Keyboard Driver (Interrupt Handler) - Acts as the interface between the physical keyboard and the operating system.It handles interrupt signals triggered by presses and converts them into encodes.
3. System-Level Message Queue - After the keyboard driver processes input, data is passed to a system-level message queue. This queue organizes input for system-wide applications and allows real-time processing by the OS.
4. Application-Level Message Queue - System messages are passed to this queue,which handles input distribution specific applications. The applications also read the key input from this queue for their user interfaces.
5. Keylogger Interception - The keylogger hooks into the system-level message queue,extracting input data before it reaches user-level applications. This placement allows it to operate stealthily,without interrupting or modifying the normal behavior of applications.
6. Data Redirection and Ex filtration- Once keystroke data is captures, it may redirected to Locl logs(e.g, .txt files in hidden folders).

RESULT DISCUSSION

To evaluate the effectiveness and stealth capability of the implemented keylogger, a series of tests were conducted in a controlled environmant using a windows-based system.

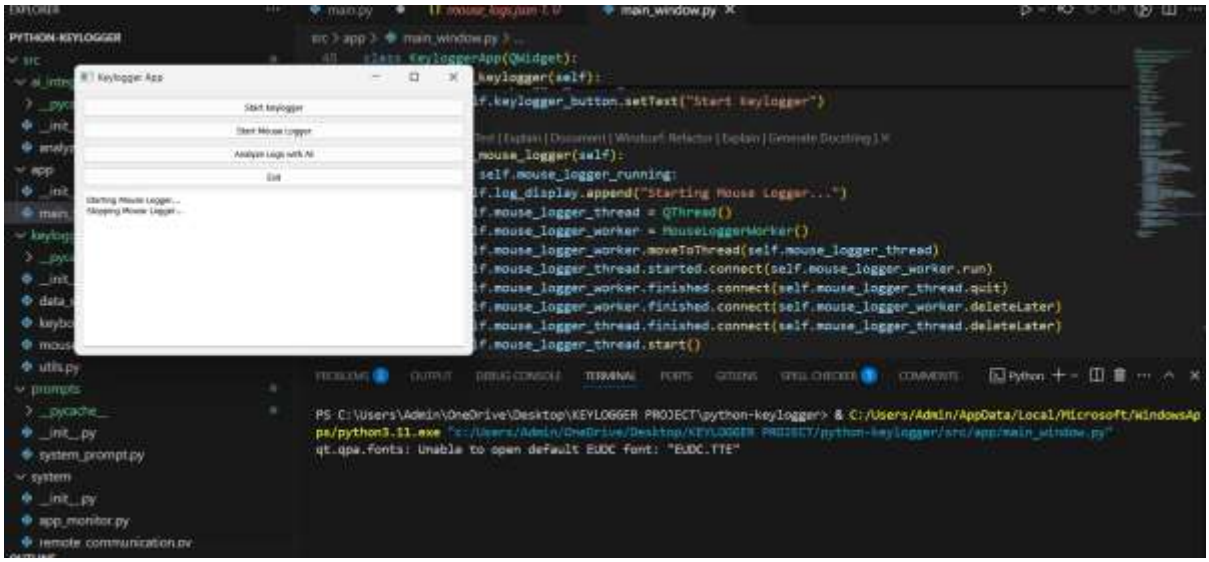

Fig.4. Mouse logging

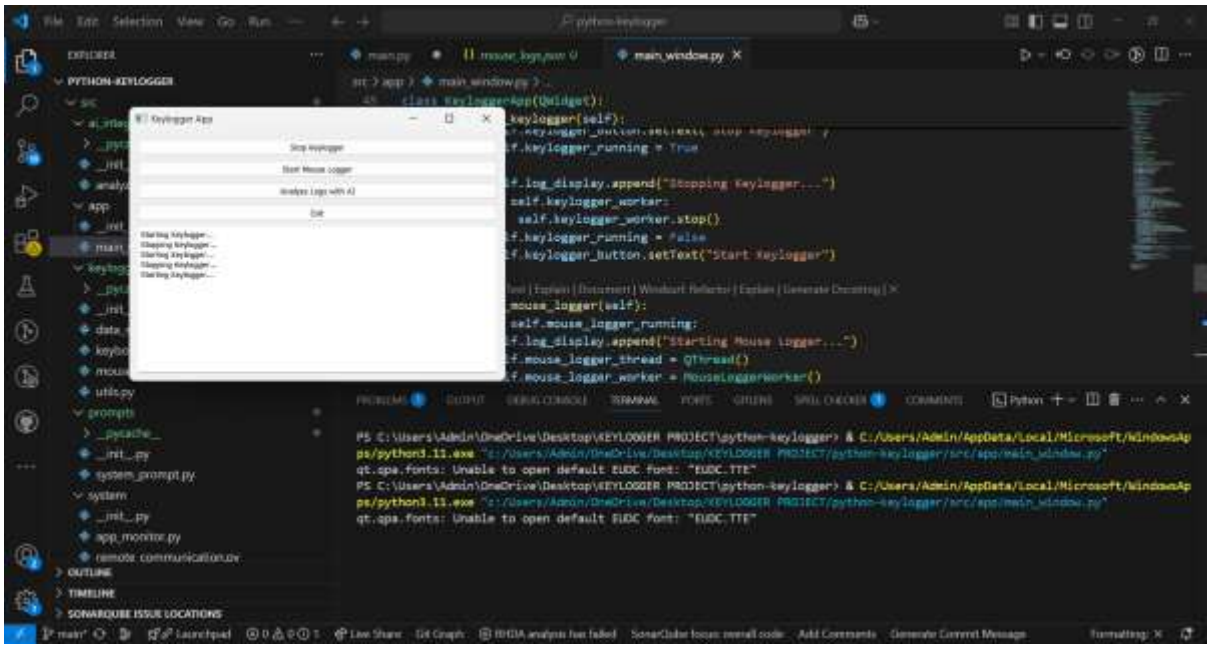

Fig.5. Keylogging Application

CONCLUSION

Keylogger forensics is a critical aspect of digital forensics that involves the identification, analysis and recovery of evidence related to keylogger attacks. By understanding the techniques used by attackers and employing advanced forensic methodologies, organisations can effectively respond to keylogger incidents and protect sensitive information.Keylogger forensics is a constantly evolving field due to the rapid advancements in technology and the increasing sophistication of cyber threats. To stay ahead of attackers, forensics analysts must continuously update their knowledge and skils, and organisations must invest in robust security measures to prevent and detect keylogger attacks. The system plays a crucial role in detecting,analyzing and mitigating keylogger threats on digital systems.By utilizing a combination of signature-based detection, behavioral analysis and AI-driven models, the system can effectively identify keylogger activity, extract relevant forensic data and generate detailed reports for incident response. The system's ability to accurately detect keyloggers, coupled with its ability to recover captured data and provide actionable insights, keyloggers, coupled with its ability to recover captured data and provide actionable insights, significantly enhances cybersecurity efforts. Furthermore, with a focus on continuous monitoring and timely mitigation actions, this forensic system aids in preventing further compromises,ensuring system integrity and safeguarding sensitive information from malicious actors. The overall accuracy and efficiency of the system make it a valuable tool for both forensic invstigators and security professionals in protecting against keylogger-based threats.

References

1. S.Moses, J.Mercado, A.Larson and D. Rowe, "Touch interface and keylogging malware,"2015 11th International Conference on Innovations in Information Technology (IIT), Dubai,2015,pp. 86- 91.doi: 10.1109/INNOVATIONS.2015.7381520
2. P Tuli, P. Sahu, "System Monitoring and Security Using Keylogger", International Journal of Computer Science and Mobile Computing,IJCSMC,Vol.2,Issue.3,March2013,pg.106-111.
3. Murugan, S; Kuppusamy, K.'System and methodology for unknown malware attack',Second IEEE

International Conference on Sustainable Energy and intelligent System(SEISCON 2011)

4. A. Solairaj. S.C.Prabanand, J.Mathalairaj, C. Prathap and L. S.Vignesh, "Keyloggera software detection techniques,"2016 10th Internatinal Conference on Intelligent Systems and Control (ISCO),Coimbatore.2016 10th,pp 1-6,doi: 10.1109/ISCO.2016.7726880..
5. Ladakis, L.Koromilas, G. Vasiliadis, et.al,"You Can Type, but You Can't Hide: A Stealthy GPU-based keylogger", EuroSec'13 April 14 2013,Prague,Czech Republic..
6. M, Dadkhak., A.Ciobotaru, et al, "An Introduction to Undetectable Keyloggers with Experimentalm Testing", International Journal of Computer Networks and Communications Security- September 2014
7. Arghire, I.'Business users targeted by Hawkeye keylogger malware'. Security Week. 28 May 2019.Accessed Jan 2020.
8. A. Bhardwaj, S. Gounder, "Keyloggers; silent cyber security weapons",2020 Network Security Volume 2020, Issue 2, February 2020, Pages 14-19.
9. Javaheri, D; Hosseinzadeh, M; Rathmani, M. 'Detection and elimination of spyware and ransomware by intercepting kernellevel system routines.' IEEE Access.
10. Sagiroglu Seref & Canbek, Gurol, (2009) Keyloggers Increasing Threats to Computer Security and Privacy Technology and Society Magazine, IEEE. 28. 10 - 17. 10. 1109/MTS.2009.934159.. Asian Journal of Convergence in Technology ISSN NO; 2350-1146 I.F-5.11 Volume VII and Issue I.
11. S. Shetty. Introduction to spyware keyloggers. www,securityfocus.com/infocus/1829, 2005..
12. Comparing the proposed virtual keyboard with QWERTY and ABC keyboards. Februsry 2020 Network Security 19 FEATYRE Volume 6, 2018. DOI: 10.1109/ACCESS.2018.2884964